REMARKS

Section 112 rejection

In response to the section 112 rejection, Applicant amends claims 1, 8, and 15 to clarify the subject matter of the invention.

Section 102 rejection

Macon[1] appears to teach a system in which a CPU **2** provides a demand address ("DADDR") to a file system **10**. The file system then compares the demand address with addresses **7a**, **7b** of pre-fetched data that have already been read from a disk **4**. If a match is found between the demand address and any of the addresses of already pre-fetched data, the pre-fetched data is retrieved from the memory **3**.

Of the addresses of pre-fetch data available in memory **3**, one is the most recently fetched pre-fetch data. This most recently read pre-fetched data is identified by the address "MRRS" in FIG. 3. The remaining pre-fetch data available in memory **3** is identified by the addresses "ENTRY" in FIG. 3. Depending on whether the demand address (DADDR) matches the most recent pre-fetch data (MRRS) or not, a length counter **16** is adjusted. This adaptively adjusted length counter **16** controls the extent to which data is read ahead during subsequent pre-fetch operations. The cited language in column 5, lines 45-61, and the accompanying FIG. 6, refer to the details of this adjustment.

It is apparent that the cited language does not teach or suggest what would occur if a demand address were to be presented during a pre-fetch operation.

Macon does appear to discuss how the control flow shown in FIG. 6 would be affected if the demand address were to be presented before completion of a pre-fetch operation.[2] However, *Macon* does not teach or suggest that the demand for data would be satisfied by retrieving data

---

[1] *Macon, Jr.*, et al., U.S. Patent 5,600,817.
[2] *Macon, Jr.*, col. 9, lines 19 et seq.

that had already been pre-fetched during the still-incomplete pre-fetch operation. Accordingly, *Macon* fails to meet the limitations of the pending independent claims.

Section 103 rejection

Dependent claims 7, 14, 21, and 42 stand rejected as obvious over the combination of *Macon* and *Cherukuri*.[3] These claims are dependent on claims that are allowable for reasons set forth above. Accordingly, these dependent claims are also allowable.

Conclusions

Now pending in this application are claims 1-42. Of these, claims 1, 8, 15, 22, 25, 28, 31, 34, 37, and 40 are independent. For reasons set forth above, these claims are now in condition for allowance.

No additional fees are believed to be due in connection with the filing of this response. However, to the extent fees are due, or if a refund is forthcoming, please adjust our deposit account 06-1050.

Attached is a marked-up version of the changes being made by the current amendment.

Respectfully submitted,

Date: Nov. 7, 2002

Faustino A. Lichauco
Reg. No. 41,942

Fish & Richardson P.C.
225 Franklin Street
Boston, Massachusetts 02110-2804
Telephone: (617) 542-5070
Facsimile: (617) 542-8906

20536063.doc

[3] *Cherukuri*, U.S. Patent 6,006,307.

Version with markings to show changes made

In the claims:

Claims 1, 8, and 15 have been amended as follows:

1. (Twice Amended) A method comprising:

initiating a reading of prefetch data in response to a request for prefetch data;

before completion of the reading of prefetch data, receiving a request for demand data; and

satisfying the request for demand data by providing at least a portion of the [~~with~~] prefetch data prior to completing the reading of all of the prefetch data.

8. (Twice Amended) An article comprising a machine-readable storage medium which stores executable instructions that cause a machine to:

initiate a read of prefetch data in response to a request for prefetch data;

before completion of the read of prefetch data, receive a request for demand data; and

satisfy the request for demand data by providing at least a portion of the [~~with~~] prefetch data prior to completing the reading of all of the prefetch data.

15. (Twice Amended) An apparatus comprising:

a memory which stores executable instructions; and

a processor which executes the instructions to:

initiate a read of prefetch data in response to a request for prefetch data;

prior to completion of the read, receive a request for demand data; and

satisfy the request for demand data by providing at least a portion of the [~~with~~] prefetch data prior to completing the reading of all of the prefetch data.